Project # 1: Finding Lane Lines on the Road

Ximing Chen

June 12, 2018

In this project, we aim to find lane lines on the road. The goals and steps of this project are the following: (i) make a pipeline that finds lane lines on the road, and (ii) reflect on the work in a written report.

1 Reflection

1.1 Pipeline

Our pipeline consisted of steps, described in details as follows:

1. We transform the colored image into a gray scale image.

2. We presented a Gaussian filter with certain kernel size (5, in this project) to prune high-frequency noises. Then, we use Canny's edge detection algorithm to identify the edges in the smoothed gray image.

3. We defined a polytopic region and prune out every pixel outside of this region; hence only keep edges within this region.

4. We performed Hough-transform on the gray-scale image containing edges in the selected region. In addition, we changed the x-y coordinate into polar coordinate to avoid infinite-slope problem. Based on Hough-transform, we retrieve the lines of interest.

5. We connect the line segments as follows. We split the polygon into two regions (i.e., left and right) by vertically cutting the image from the middle. Subsequently, each of the two images should contain only one red line (or a few line segments of similar slope). Then, we define a line ℓ according to the slopes retrieved from the detected line segments. Finally, we obtain the tuple of points (x_1, y_1) and (x_2, y_2) that lie on the boundaries of the polygon as well as ℓ. By connecting (x_1, y_1) to (x_2, y_2), we obtain the desired lines.

6. Finally, we overlay the colored lanes on top of the original image.

1.2 Shortcomings

There are a few short comings listed as follows:

- The polygon is hard-coded. In other words, it only work for lanes that are straight enough. If the lane in practice is curving, then our method will not align with the lanes. Furthermore, if the a car changes lane, we may only detect a single lane and label other irrelevant information as lanes incorrectly.
- Deciding whether certain pixels form a line in Hough transform is purely based on heuristics counts as of now. The shortcoming due to this is that other cars on the road may be labelled as lines (e.g. a truck or any objects whose shape is similar to a rectangle of line).
- The detection is not robust with respect to the environment. For example, when there is a car in front, the Canny edge detection algorithm will also find the edges corresponding to the shape of the car. As a result, the approach we adopted to connect the line segments will result in error.
- We process the video frame-by-frame. As a result, it may not be possible to perform real-time lane detection.

1.3 Possible Improvements

- The idea of using a region-of-interest is to remove irrelevant information. Thus, it is possible to use other filters to remove other irrelevant information and hence keeping the pixels that represent lanes. For example, we may first remove cars of certain color (e.g., black, green since those colors will never be the lane color).
- In our current write-up, we adopted a few end points for line segment extension. However, it would be better to incorporate more data-points. For example, we may consider retrieving the slopes from the Hough space, or perform a linear regression among the points that corresponds to lines obtained from Hough-transform.